# A Tour of Constrained Tensor Canonical Polyadic Decomposition

Jérémy E Cohen, Konstantin Usevich, Pierre Comon

▶ **To cite this version:**

Jérémy E Cohen, Konstantin Usevich, Pierre Comon. A Tour of Constrained Tensor Canonical Polyadic Decomposition. [Research Report] GIPSA-lab. 2016. hal-01311795

**HAL Id: hal-01311795**
**https://hal.science/hal-01311795**

Submitted on 4 May 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A Tour of Constrained Tensor Canonical Polyadic Decomposition

Jérémy Emile Cohen*, Konstantin Usevich and Pierre Comon, *Fellow, IEEE*

***Abstract*—This paper surveys the use of constraints in tensor decomposition models. Constrained tensor decompositions have been extensively applied to chemometrics and array processing, but there is a growing interest in understanding these methods independently of the application of interest. We suggest a formalism that unifies various instances of constrained tensor decomposition, while shedding light on some possible extensions of existing methods.**

***Index Terms*—Tensor, Multiway analysis, Constrained optimization, Coupled decompositions, Data fusion, Compression, Big Data, PARALIND.**

## INTRODUCTION

In the recent years, a great number of works has dealt with theoretical and practical aspects of multiway data mining, which studies data obtained from simultaneous measurements at (often three) different modalities. That is, data are contained in a cube of measurements, which is called the data array or data tensor. Tensors have been used extensively in chemometrics [1], [2], neural imaging [3]–[5], antenna array processing [6], [7], fast matrix computations [8], [9], statistics [10], differential equations [11], hyperspectral image processing [12] among others. In those applications, tensors can be decomposed according to a multilinear model known as Canonical Polyadic decomposition (CP) (also referred to as PARAFAC). This decomposition infers the relations between all modalities, as opposed to standard data mining in which data blocks would be unfolded into matrices without accounting for the multilinear structure of the underlying model.

In particular, the CP decomposition model gained in popularity since, without further constraints than the model itself, parameters can be uniquely recovered under mild conditions on the decomposed tensor [13]. Since identifiability does not need to be restored, it could thus seem unnecessary to add constraints to the CP decomposition model, in contrast with PCA or Tucker models for example [14] that arbitrarily imposes orthogonality constraints on the parameters. However, adding meaningful constraints to tensor CP decomposition models turns out to be of crucial importance depending on the application at hand. Firstly, some constraints on the output of tensor decomposition help to interpret these outputs in a physical sense. For example, non-negativity constraints are often used when mining data stemming from fluorescence spectroscopy, since it makes resulted estimated spectra and concentrations physically intepretable [2]. This means that constraining the CP decomposition is a data-driven way to build tensor decomposition models. Secondly, constraining the set of parameters of a model is known to potentially decrease estimation error and can restore identifiability of an approximate problem. For instance, sparsity constraints have been hugely successful in machine learning since it makes dictionary learning a well-posed problem. For tensors, non-negativity constraints have been shown to make the approximate CP decomposition problem well-posed [15].

The authors are with GIPSA-Lab, CNRS, University Grenoble Alpes 38000 Grenoble, France (e-mail: firstname.lastname@gipsa-lab.grenoble-inp.fr). This research was supported by the ERC Grant AdG-2013-320594 "DECODA".

Manuscript received XX, 2016; revised XX, 2016.

## OUTLINE AND CONTRIBUTIONS

This paper surveys constrained CP decompositions within a unifying framework, suggesting some research topics and proposing increments to existing models along the way. It differs from recently published surveys [16], [17] since the focus is here on physically interpretable models design and giving a wide picture of constrained tensor decompositions. The first section defines the CP decomposition as well as some core concepts of tensor algebra. Simple yet widely used constraints are then discussed. More complex interactions between constraints are further exposed in the section devoted to Applications. Finally, some algorithmic issues are discussed in the last section.

## NOTATION AND VOCABULARY

Among various notation habits in the multiway array processing community, we choose to follow notation from [18], [19], as presented in Tables I and II. We call a third-order real $K \times L \times M$ tensor $\boldsymbol{\mathcal{T}}$ a vector from a tensor space $\mathbb{R}^K \otimes \mathbb{R}^L \otimes \mathbb{R}^M$. Since this tensor space is isomorphic to $\mathbb{R}^{K \times L \times M}$, third-order tensors and three-way arrays are often confused, and unnecessarily cumbersome notations are consequently used. However the array space is (only) one instance of a tensor product space and tensor product notations can be used instead of array-specific notations. Here, whenever dealing with data array, *i.e.*, when studied vectors are indeed in $\mathbb{R}^{K \times L \times M}$, the tensor product of vectors in $\mathbb{R}^K$, $\mathbb{R}^L$ and $\mathbb{R}^M$ can be cast as the outer product. In Table II, some useful properties of multilinear operators acting on tensors are specified. These operators also form a tensor space. More properties on multilinear operators can be found in [19].

Given a $K \times L \times M$ tensor $\boldsymbol{\mathcal{T}}$, its CP decomposition of rank $R$ can be written as follows:

$$\boldsymbol{\mathcal{T}} = \sum_{r=1}^{R} \boldsymbol{\mathcal{D}}_r \tag{1}$$

where $\boldsymbol{\mathcal{D}}_r$ are decomposable tensors of the form $\boldsymbol{\mathcal{D}}_r = \boldsymbol{a}_r \otimes \boldsymbol{b}_r \otimes \boldsymbol{c}_r$. The rank of $\boldsymbol{\mathcal{T}}$ is the minimal value of $R$ such that (1) holds exactly. Finding the CP decomposition of a third-order tensor means finding rank-1 tensors $\boldsymbol{\mathcal{D}}_r$. Yet, each tensor $\boldsymbol{\mathcal{D}}_r$ may be defined by three vectors $\boldsymbol{a}_r$, $\boldsymbol{b}_r$ and $\boldsymbol{c}_r$, only up to two scaling ambiguities; in fact, $\boldsymbol{a}_r \otimes \boldsymbol{b}_r \otimes \boldsymbol{c}_r = \alpha\boldsymbol{a}_r \otimes \beta\boldsymbol{b}_r \otimes \boldsymbol{c}_r/\alpha\beta$, $\forall \alpha, \beta \neq 0$.

Next, it is often convenient to store these vectors in matrices as $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_R]$, $\boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_R]$ and $\boldsymbol{C} = [\boldsymbol{c}_1, \ldots, \boldsymbol{c}_R]$. This leads to a convenient writing:

$$\boldsymbol{\mathcal{T}} = (\boldsymbol{A} \otimes \boldsymbol{B} \otimes \boldsymbol{C})\, \boldsymbol{\mathcal{I}}_R \tag{2}$$

where $\boldsymbol{A} \in \mathbb{R}^{K \times R}$, $\boldsymbol{B} \in \mathbb{R}^{L \times R}$ and $\boldsymbol{C} \in \mathbb{R}^{M \times R}$ are called factor matrices, and $\boldsymbol{\mathcal{I}}_R$ is a diagonal core tensor with only ones on the diagonal. Model (2) now contains $2R$ scaling indeterminacies (whereas definition (1) did not contain any).

Conditions on the dimensions of the tensor and rank of the decomposition are given in the literature [13], [20], [21] to ensure uniqueness of the factors in an unconstrained CP model, but only when noise is absent. They will hence not be introduced here since most applications of constrained CP must consider noise.

| |
|---|
| $\mathcal{E} \otimes \mathcal{F}$: tensor product space, linear space mapped by $\otimes$ from $\mathcal{E} \times \mathcal{F}$. |
| $\boldsymbol{a} \otimes \boldsymbol{b}$: tensor product of two vectors, *i.e.* an element of $\mathcal{E} \otimes \mathcal{F}$, can be understood as an outer product of vectors if the tensor space is an array space. [18] |
| $\boldsymbol{A} \odot \boldsymbol{B}$: Khatri-Rao (columnwise Kronecker) product of matrices [22] |
| $\boldsymbol{A} \boxdot \boldsymbol{B}$: Hadamard product of matrices, *i.e.*, element-wise product [22] |

TABLE I
BASIC DEFINITIONS FROM LINEAR ALGEBRA

## CONSTRAINED CP DECOMPOSITION

### *General Model*

There are two ways to understand the CP model. It can be understood as a pure mathematical problem where factors bear no meaning with regard to physical quantities, or it can be seen as a blind source separation model. In the latter, factors can be interpreted as sources and coefficients, meaning that the numerical values in the factors should be interpretable physically. This also means that in many applications, *a priori* information on the factors is available. For example in chemometrics, factors may refer to spectra and concentrations [1], which are obviously non-negative. More about CP decomposition applied to chemometrics is discussed in Example 2.

| |
|---|
| $\boldsymbol{U} \otimes \boldsymbol{V} \otimes \boldsymbol{W}$ : an operator acting on a third order tensor. |
| $(\boldsymbol{U} \otimes \boldsymbol{V} \otimes \boldsymbol{W})\, \boldsymbol{\mathcal{T}}$ : application of $\boldsymbol{U}$ on the first mode, $\boldsymbol{V}$ on the second mode, and $\boldsymbol{W}$ on the third mode, also noted $\boldsymbol{\mathcal{T}} \bullet_1 \boldsymbol{U} \bullet_2 \boldsymbol{V} \bullet_3 \boldsymbol{W}$. |
| $(\boldsymbol{U} \otimes \boldsymbol{V})\, (\boldsymbol{a} \otimes \boldsymbol{b}) = \boldsymbol{U}\boldsymbol{a} \otimes \boldsymbol{V}\boldsymbol{b}$. |

TABLE II
SOME DEFINITIONS AND PROPERTIES OF MULTILINEAR OPERATORS

This calls for a generalization of the CP decomposition to include potential constraints stemming from *a priori* knowledge into the decomposition model and noise distribution. Then the following general constrained CP decomposition model is obtained:

$$\begin{cases} \boldsymbol{\mathcal{T}} = (\boldsymbol{A} \otimes \boldsymbol{B} \otimes \boldsymbol{C})\, \boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}} \\ \boldsymbol{A} \in \mathcal{S}_A,\ \boldsymbol{B} \in \mathcal{S}_B,\ \boldsymbol{C} \in \mathcal{S}_C \end{cases} \tag{3}$$

where $\mathcal{S}_A$ (resp. $\mathcal{S}_B$ and $\mathcal{S}_C$) can be any *constraint space* included in $\mathbb{R}^{K \times R}$ (resp. $\mathbb{R}^{L \times R}$ and $\mathbb{R}^{M \times R}$), and $\boldsymbol{\mathcal{E}}$ is a Gaussian noise on every coefficient of $\boldsymbol{\mathcal{T}}$. Note that (3) is an approximate constrained decomposition model since noise is considered.

The nature of (3) depends heavily on the constraint spaces defined above. In the following, we present some examples of constraint spaces that are being considered in the literature. For the sake of simplicity, we suppose from now on that only the factor $\boldsymbol{C}$ is constrained, although the following discussions are unaltered if all factors are constrained. Only the paragraph dealing with compression with constraints features constraints on all three modes since compressing only dimension typically makes little sense.

### *Linear constraints*

A first approach to constrained CP is to consider only linear constraints on columns of factors, or constraints that can be linearized in this way. Then the constrained factor $\boldsymbol{C}$ can be expressed with a spanning family $\boldsymbol{W}$ of the linear constraint space,

$$\boldsymbol{C} \in \mathrm{Span}\{\boldsymbol{W}\} \subseteq \mathcal{S}_C. \tag{4}$$

This yields the following model :

$$\begin{cases} \boldsymbol{\mathcal{T}} = (\boldsymbol{A} \otimes \boldsymbol{B} \otimes \boldsymbol{C})\, \boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}} \\ \boldsymbol{C} = \boldsymbol{W}\boldsymbol{C}_{\boldsymbol{c}} \end{cases} \tag{5}$$

where $\boldsymbol{C}_{\boldsymbol{c}}$ is a matrix of coefficients, of size $R_3 \times R$, $R_3 < M$.

Linear constraints on columns of $\boldsymbol{C}$ are useful to impose that each component in the third mode belongs to a certain class of functions and was first studied under the name CANDELIND [23]. In chemometrics for instance, bases of splines have been used to impose smoothness on one factor [24]. Such examples of linear constraints are however scarce in the literature. In the section devoted to algorithms, we show that linear constraints with known spanning families of full column rank can be handled through already existing algorithms with some modification of the noise distribution, so that there is no technical issue to their use within already known methods.

A common subclass of linearly constrained CP decomposition is obtained when the mixing matrix $\boldsymbol{W} \in \mathbb{R}^{M \times R_3}$ is tall and orthonormal. Such constraints allow for compression by simply projecting all columns of $\boldsymbol{C}$ onto a smaller subspace spanned by matrix $\boldsymbol{W}$ in which row correlation is decreased:

$$\begin{aligned} \boldsymbol{\mathcal{T}} &= (\boldsymbol{I} \otimes \boldsymbol{I} \otimes \boldsymbol{W})\, \boldsymbol{\mathcal{G}} + \boldsymbol{\mathcal{E}} && (6) \\ &= (\boldsymbol{I} \otimes \boldsymbol{I} \otimes \boldsymbol{W})\, (\boldsymbol{A} \otimes \boldsymbol{B} \otimes \boldsymbol{C}_{\boldsymbol{c}})\, \boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}} && (7) \\ &= (\boldsymbol{A} \otimes \boldsymbol{B} \otimes \boldsymbol{W}\boldsymbol{C}_{\boldsymbol{c}})\, \boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}} && (8) \end{aligned}$$

where $\boldsymbol{\mathcal{G}}$ is a small $K \times L \times R_3$ core tensor, generally not diagonal, and $\boldsymbol{C_c}$ is the $R_3 \times R$ third mode factor of $\boldsymbol{\mathcal{G}}$. Also $\boldsymbol{\mathcal{E}}$ remains independent and identically distributed. Only $\boldsymbol{\mathcal{G}}$ needs to be decomposed in an unconstrained fashion to obtain the linearly constrained CP decomposition of $\boldsymbol{\mathcal{T}}$. If $\boldsymbol{W}$ were not tall, *i.e.* if $R_3$ were greater than $M$, then the constraint space would be the whole space of definition of $\boldsymbol{C}$, and the constraint would be unuseful since always verified. Constraint (5) is often induced by the rank constraint described in the next subsection.

In some applications like antenna array processing, factors in the CP decomposition may have Hankel, Toeplitz or another particular structure [25], [26]. Since these structured matrices form a linear space, taking into account the structure amounts to linearly constrained CP as described in (5). In practice, for constraint spaces of small dimension, specific finite algorithms may be devised [25], [26], possibly to initialize iterative algorithms.

On another topic, let us consider the case where $\boldsymbol{C}$ is a multivariate random variable given by a linear additive model with noise $\boldsymbol{\nu}$ following some given distribution.

$$\boldsymbol{C} = \boldsymbol{W}\boldsymbol{C_c} + \boldsymbol{\nu}. \tag{9}$$

The CP decomposition constrained by this stochastic equation designated as approximate linear constraints has been very little studied. However, if $\boldsymbol{W}$ and $\boldsymbol{C_c}$ need to be estimated given some *a priori* knowledge, a flexible constraint may prevent over-fitting the constraint space, which will not be correctly defined because of estimation errors. Moreover, it also models factor $\boldsymbol{C}$ being close to the span of $\boldsymbol{W}$, the distance being some prior probability distribution more or less known. Some work on approximate linear constraints can be found in [27], [28] in the context of data fusion.

*Rank constraints*

A rank constraint can be cast on the rank of factor $\boldsymbol{C}$, by setting

$$\mathcal{S}_C = \{\boldsymbol{X} \in \mathbb{R}^{M \times R} \mid \operatorname{rank}\{\boldsymbol{X}\} < \min(M, R)\}. \tag{10}$$

This constraint is difficult to take into account as is, since $\mathcal{S}_C$ is not a linear space. Yet it can be simplified as it implies the existence of two orthogonal bases, defined by the columns of two matrices $\boldsymbol{W}$ and $\boldsymbol{H}$, respectively in $\mathbb{R}^{M \times \operatorname{rank}\{\boldsymbol{X}\}}$ and $\mathbb{R}^{\operatorname{rank}\{\boldsymbol{X}\} \times R}$ so that $\boldsymbol{C} = \boldsymbol{W}\boldsymbol{C_c}\boldsymbol{H}$, with $\boldsymbol{H}^\mathsf{T}\boldsymbol{H} = \boldsymbol{I}$ and $\boldsymbol{W}^\mathsf{T}\boldsymbol{W} = \boldsymbol{I}$. In other words constraint (10) can be linearized. Since in (10), $\operatorname{rank}\{\boldsymbol{C}\}$ is strictly lower than $M$, at least there exist $\boldsymbol{W}$ and some matrix $\boldsymbol{N}$ so that $\boldsymbol{C} = \boldsymbol{W}\boldsymbol{N}$ where $\boldsymbol{W}$ is tall and orthogonal. Thus under a low rank constraint on $\boldsymbol{C}$, compression can be applied if $\boldsymbol{W}$ can be learnt. The collinearity among columns induced by a fat $\boldsymbol{H}$ is discussed in the next section through the PARALIND model.

Another constraint set is often used for $\boldsymbol{C}$. A large majority of CP decompositions are subject to a low rank constraint that is easier to express,

$$\mathcal{S}_C = \{\boldsymbol{X} \in \mathbb{R}^{M \times R} \mid \operatorname{rank}\{\boldsymbol{X}\} \leqslant R \ll M\}. \tag{11}$$

If columns of $\boldsymbol{C}$ are independent, the column rank of $\boldsymbol{C}$ is equal to $R$. Moreover $\boldsymbol{C} \in \mathcal{S}_C$ will have linearly dependent rows. If (11) is imposed on all factors $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$, the set of constraints defines the tensor low-rank constraint.

This is at the very core of data mining that from high rank noisy data should emerge a small number $R$ of informative components. Thus tensor low-rank constraint is almost a must-have for any tensor-based analysis, including classical two-way (*i.e.* matrix) data analysis [29]. In practice under tensor low-rank constraint, rows of all factors whose dimension is larger than the tensor rank are generically linearly dependent, so that again compression can be performed on all modes of the tensor.

*Nonlinear constraints*

Many applications require nonlinear constraints. A frequently encountered one is non-negativity, essential in chemometrics or hyperspectral imaging for instance. With a non-negativity constraint on the third mode, the model becomes

$$\begin{cases} \boldsymbol{\mathcal{T}} = (\boldsymbol{A} \otimes \boldsymbol{B} \otimes \boldsymbol{C})\,\boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}} \\ \boldsymbol{C} \in \mathbb{R}_+^{M \times R} \end{cases}. \tag{12}$$

This has been used extensively whenever factors bear physical interpretation in chemometrics [2], [30], but also in neuroscience [31], [32]. It is also especially popular for second order tensors under the Non-negative Matrix Factorization (NMF) acronym [33], [34], since this constraint for matrices often restores identifiability of the factors.

For higher order tensors as well, non-negativity constraints ensure that a best low rank approximation exists [35] for all norms. Remember that this is the only constrained approximate CP decomposition where such a guarantee exists yet.

Another common non-linear constraint is the $\ell_0$ sparsity constraint [36], [37] :

$$\begin{cases} \boldsymbol{\mathcal{T}} = (\boldsymbol{A} \otimes \boldsymbol{B} \otimes \boldsymbol{C})\,\boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}} \\ \delta_1 \leqslant \ell_0(\boldsymbol{C}) \leqslant \delta_2 \end{cases} \tag{13}$$

for some integers $\delta_1 \leqslant \delta_2$ and the $\ell_0$ *pseudo-norm* can be taken rowwise, columnwise or on all elements. This means that each of the $R$ components in the CP decomposition contributes only sparsely to tensor $\boldsymbol{\mathcal{T}}$ on the third mode. Sparsity constraints are meaningful when at least one of the factor is understood as a dictionary, and factor $\boldsymbol{C}$ has a small number of nonzero coefficients. To our knowledge, it has not been studied yet whether sparsity constraints have any impact on the existence and uniqueness of a best approximate CP decomposition.

The $\ell_1$ norm can be used in a similar fashion, as a relaxation of the $\ell_0$ norm for the optimization routine, or as a constraint in itself. For instance, for $\boldsymbol{C}$ stemming from the decomposition of hyperspectral data as presented in Example 1, rows of $\boldsymbol{C}$ should sum to one which leads to $\ell_1(\boldsymbol{C}) = 1$ row-wise. The $\ell_1$ norm also serves as a regularizer for dictionary learning with matrices. Sparsity for dictionary-based CP decomposition is explored in the next section devoted to Applications.

**Example 1:** CP decomposition can be used to process multispectral and hyperspectral images, under the condition that a third diversity is provided (time, angle, shift ...) [12]. Under multilinearity hypothesis, an image $\boldsymbol{\mathcal{X}}$ is decomposed into a matrix of abundances $\boldsymbol{A}$, which contains the proportion of each component within each pixel, a matrix of spectra $\boldsymbol{B}$, which contains a reflectance spectrum for each material, and a third factor matrix $\boldsymbol{C}$ containing for example temporal signatures of each source. Each factor is non-negative, and the abundances in $\boldsymbol{A}$ can be modeled to sum to one in each row since they are percentages of the total contribution of each pixel. Moreover, a dictionary of spectra can be provided. In Figure 1, we use data from [12] to compare unconstrained decomposition and constrained dictionary-based CP decomposition. The decomposition model and algorithm are described in Appendix B. When the dictionary is provided in the decomposition model, identification of the recovered spectral signatures is automatic, and all spectra can be interpreted. On the other hand, when the dictionary-based CP is not used, identification of spectra in Figure 1 (upper left plot) may not be possible.

Other constraints like smoothness or unimodality have been studied in the literature [2], but for concision purposes and since they have often not been applied in the literature, these constraints will not be presented here.

## APPLICATIONS

In what follows, we study combinations of linear constraints with non-linear constraints under the tensor low-rank constraint. Multimodal tensorial data models are also described as constrained CP. We also discuss dictionary-based CP decomposition. In this section the rank of the decomposition is considered small with respect to tensor dimensions as in (11).

### *Collinearity in Factors*

When decomposing some tensorial data with the CP model, it can occur that some factors should have different number of components (cf. Example 2). Since the rank chosen for the decomposition fixes the number of components for all factors, this means that collinearity in the columns of factors can be encountered. In other words, we study the following constrained CP decomposition :

$$\begin{cases} \boldsymbol{\mathcal{T}} = (\boldsymbol{A} \otimes \boldsymbol{B} \otimes \boldsymbol{C})\, \boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}} \\ rank(\boldsymbol{C}) < R \ll M \end{cases} \tag{14}$$

where the constraint amounts to (10). Here we focus on the implication of the rank constraint on the column space of $\boldsymbol{C}$.

**Example 2:** Let us consider a naive experiment where factors should have different column ranks. Take a solution with three fluorescent components at different concentrations. Using a spectrophotometer, a mixture of the three emission spectra and the three excitation spectra can be acquired in the form of a matrix. Now by adding a bit of the third component and diluting the whole solution, running the experiment again gives another matrix. Repeating this several times will result in a collection of matrices, our measurement tensor, where the two first concentration profiles are collinear. Thus the column rank of the factor $\boldsymbol{C}$ related to concentration will be two, even though the rank of the tensor will be three. A better model would thus use a reduced set $\boldsymbol{C_c}$ of parameters by setting

$$\boldsymbol{C} = \boldsymbol{C_c} \begin{bmatrix} 1 & \lambda & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where $\lambda$ is the ratio between the two constant concentration profiles. This model is called PARALIND [38], and is discussed below.

Having collinear columns in the CP decomposition is not a problem in terms of modeling, but it raises serious issues with respect to conditioning of the underlying optimization problem. For this reason, two approaches have been proposed to reformulate (14), namely $(L_r, L_r, 1)$ Block Term Decomposition and PARALIND [4], [38]–[40]. It is worth noting that these two models are equivalent, which has remained widely unnoticed.

As explained in the previous section, a rank constraint on columns of $\boldsymbol{C}$ can be relaxed into a linear constraint :

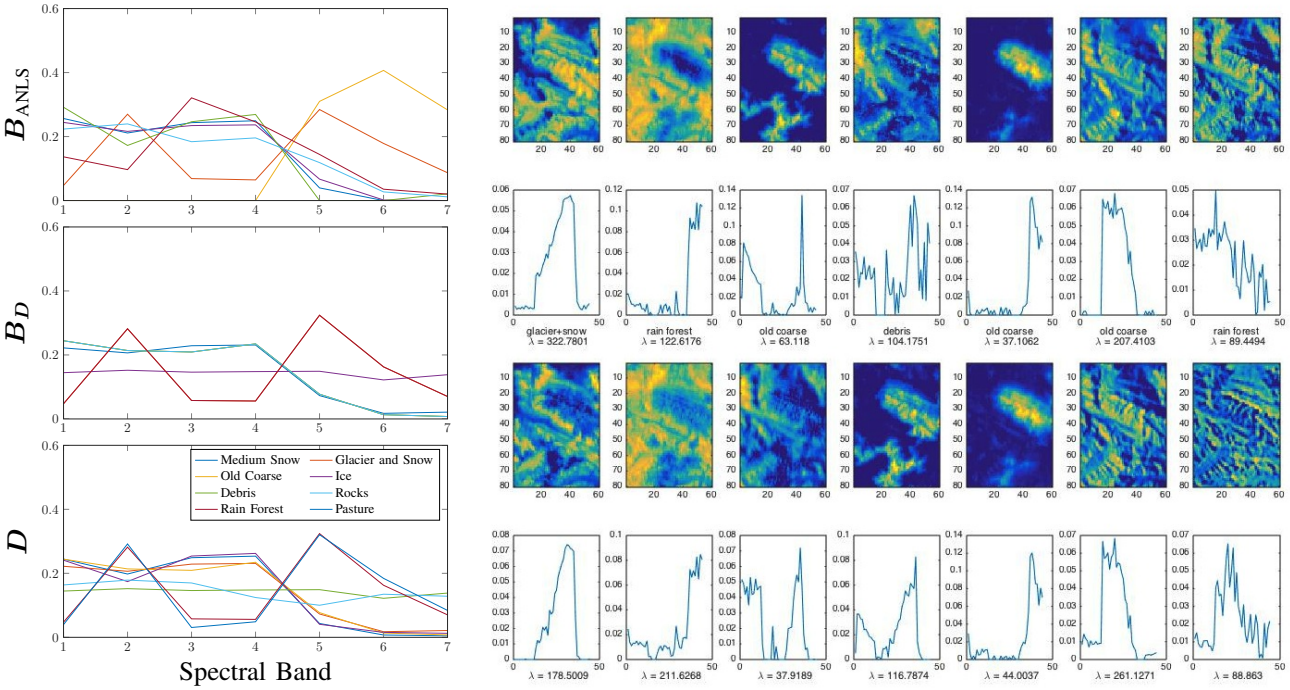
$$\boldsymbol{C} = \boldsymbol{C_c}\boldsymbol{H} \tag{15}$$

where $\boldsymbol{C_c} \in \mathbb{R}^{M \times r}$ with $r < R$ and $\boldsymbol{H}$ in a mixing matrix acting on the columns of $\boldsymbol{C_c}$ to account for collinearity between columns of $\boldsymbol{C}$. Typically $\boldsymbol{H}$ is flat and full row rank, see Example 2.

If $\boldsymbol{H}$ is known, then it simply needs to be included in the gradient computation (26) when estimating $\boldsymbol{C_c}$. However, in most cases $\boldsymbol{H}$ will also be unknown, and may even be the most important set of parameters to estimate from the data. Without *a priori* knowledge, $\boldsymbol{H}$ is not identifiable, since many couples $(\boldsymbol{H}, \boldsymbol{C_c})$ can yield a low column rank matrix $\boldsymbol{C}$ [41]. To provide a meaningful estimation of this mixing matrix, the S-PARALIND model was developed [42]:

$$\begin{cases} \boldsymbol{\mathcal{T}} = (\boldsymbol{A} \otimes \boldsymbol{B} \otimes \boldsymbol{C})\, \boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}} \\ \boldsymbol{C} = \boldsymbol{C_c}\boldsymbol{H} \\ \ell_0(\boldsymbol{H}) \leqslant \delta \end{cases} \tag{16}$$

where the $\ell_0$ pseudo-norm is taken on all the coefficients of $\boldsymbol{H}$. It is shown that, contrary to the PARALIND model where $\boldsymbol{H}$ is not identifiable, adding a sparsity constraint on the mixing matrix yields a condition on obtaining the sparsest $\boldsymbol{H}$ [41]. In other words, S-PARALIND is easier to interpret since the sparsest possible mixing matrix is selected. However, this is balanced by an increased complexity of the underlying optimization problem.




(a) Recovered and Dictionary Spectra

(b) Abundances and Temporal signatures

Fig. 1. Recovered factors from non-negative CP decomposition versus dictionary-based non-negative CP decomposition.

*Dictionaries*

Because of its similarity to matrix factorization, the CP decomposition is sometimes interpreted as a dictionary learning model [36], [43]. Indeed, for two-way arrays, the CP decomposition can be written has $\boldsymbol{T} = \boldsymbol{D}\boldsymbol{S}$ where $\boldsymbol{D}$ is the dictionary and $\boldsymbol{S}$ stands for the scores.

However, we see strong differences between low rank unconstrained CP decomposition factors and dictionaries obtained in dictionary learning methods. Firstly, for a low rank CP model, factors have at most $R$ columns and as stated earlier, $R$ is usually chosen small with respect to the dimensions since it corresponds to the number of components of interest in the data. On the other hand, dictionaries can be over-complete families of atoms, so that usually the number of atoms is far greater than their size. Secondly, when considering tensorial data, each mode stems from a different modality. A dictionary is often related to one modality – for instance a library of spectra when dealing with spectral images – so that dictionaries may help identifying each modality separately, *i.e.* should help recovering the factors instead of being the factors themselves.

Hence we are led to consider a new model for dictionary-based CP decomposition:

$$\begin{cases} \boldsymbol{\mathcal{T}} = (\boldsymbol{A} \otimes \boldsymbol{B} \otimes \boldsymbol{C})\, \boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}} \\ \boldsymbol{C} = \boldsymbol{D}\boldsymbol{C_c} \\ \ell_0(\boldsymbol{C_c}) \leqslant \delta \end{cases} \tag{17}$$

where $\boldsymbol{D}$ is an over-complete dictionary in $\mathbb{R}^{M \times p}$ and $p \gg R$.

This model was first suggested in [44] in the specific context of harmonic retrieval with the $\ell_0$ pseudo-norm taken column-wise and $\delta$ is set to 1. It is similar to S-PARALIND (16), except that the number of columns in $\boldsymbol{D}$ is greater than $R$. Moreover, $\boldsymbol{D}$ might be provided as an *a priori* known basis for decomposing factor $\boldsymbol{C}$. Dictionary-based CP decomposition can be understood also as a data fusion model between tensor data and a known sparse representation basis of $\boldsymbol{C}$. Using dictionaries can improve interpretability in applied CP decompositions. To illustrate this, an example of over-complete dictionary-based CP decomposition for hyperspectral data is provided in Example 1 and Figure 1, but most details are reported in Appendix B.

*Compression with constraints*

In the previous section, compression for tensors following a low tensor rank CP model has been described in (7) and (11). It yields that decomposing a $K \times L \times M$ tensor $\boldsymbol{\mathcal{T}}$ into a CP of rank $R$ is equivalent in the least squares sense to decomposing a smaller $R \times R \times R$ tensor $\boldsymbol{\mathcal{G}}$ provided a basis for each mode $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{W}$ is known. In other words, for a Gaussian noise $\boldsymbol{\mathcal{E}}$, model (11) is equivalent to

$$\begin{cases} \boldsymbol{\mathcal{G}} = (\boldsymbol{A_c} \otimes \boldsymbol{B_c} \otimes \boldsymbol{C_c})\, \boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}} \\ \boldsymbol{A} = \boldsymbol{U}\boldsymbol{A_c},\ \boldsymbol{B} = \boldsymbol{V}\boldsymbol{B_c},\ \boldsymbol{C} = \boldsymbol{W}\boldsymbol{C_c} \end{cases}. \tag{18}$$

Bases $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{W}$ can be estimated through the Tucker decomposition with orthogonality constraints [14], or the Maximum Likelihood High Order Singular Value Decomposition, which itself is well approximated by computing the SVD of tensor $\boldsymbol{\mathcal{T}}$ unfolded along each mode [45], [46]. Orthogonality is often imposed so that $\boldsymbol{\mathcal{E}}$ remains i.i.d., but only left invertibility of $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{W}$ is required to perform compression.

Sometimes it is necessary to choose a dimension higher than $R$ for the compressed space to account for estimation errors in the column basis $\boldsymbol{W}$ [46].

The tricky part with compression is to include other non-linear constraints. If constraints apply on the uncompressed factor, then after compression, parameters are optimized and constrained in two different spaces. This issue is discussed in [47] where an alternating algorithm is designed, going back and forth between compressed and uncompressed spaces. When non-linear constraints are imposed on $\boldsymbol{C}$, the following model is obtained:

$$\begin{cases} \boldsymbol{\mathcal{G}} = (\boldsymbol{A_c} \otimes \boldsymbol{B_c} \otimes \boldsymbol{C_c})\, \boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}} \\ \boldsymbol{A} = \boldsymbol{U}\boldsymbol{A_c},\ \boldsymbol{B} = \boldsymbol{V}\boldsymbol{B_c},\ \boldsymbol{C} = \boldsymbol{W}\boldsymbol{C_c} \\ \boldsymbol{C} \in \mathcal{S}_c \end{cases} . \tag{19}$$

We want to emphasize here that any property on existence and uniqueness achieved by adding constraints to the CP decomposition is challenged when using compression. For instance, when compressing a non-negative tensor, existence of a best non-negative low-rank decomposition is not directly inherited from [35]. Using the Perron-Frobenius theorem, we prove in Appendix A the existence of a best low rank approximation for compressed non-negative tensors.

### *Joint Decompositions*

When acquiring multiple multiway data sets from various modalities, it may not be wise to consider a single high order tensor of measurements. Rather, when these modalities are linked through factors in the CP decomposition, using coupled models can again improve interpretability and decrease estimation error on the linked factor. An example of EEG and MEG coupling is provided in Sidebar 3.

Coupled tensor decompositions have first been studied by Harshmann [48], more recent models are studied by Acar et alterae [49]. The simplest form of coupled CP decomposition can be formalized as

$$\begin{cases} \boldsymbol{\mathcal{T}}_i = (\boldsymbol{A}_i \otimes \boldsymbol{B}_i \otimes \boldsymbol{C}_i)\, \boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}}_i \\ \boldsymbol{C}_i = \boldsymbol{C} \quad \forall i \leqslant n \end{cases} \tag{20}$$

where $n$ is the number of coupled data sets. Note that here the scaling ambiguity may hinder the model design and should be pulled in one of the factors, as mentioned in [28].

Many refinements of this model can be found in the literature. Some focus on coupling only a subset of the components by constraining the number of coupled sources to be low in an unsupervised fashion [49], others generalize the coupling relationship to account for transformations and error in the coupling constraint [27]. Theoretical results on identifiability when noise is absent are provided in [50], and the Cramér-Rao bound for (20) are computed in [28].

Interestingly, joint decompositions can be used in combination with linear constraints when the coupling relationship occurs in a transformed space. For instance, factors may be coupled though their derivative [51] or through a learnt transformed basis, the latter being the core idea behind PARAFAC2, a refined model stemming from the CP decomposition [52], [53]. This yields:

$$\begin{cases} \boldsymbol{\mathcal{T}}_i = (\boldsymbol{A}_i \otimes \boldsymbol{B}_i \otimes \boldsymbol{C}_i)\, \boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}}_i \\ \boldsymbol{C}_i = \boldsymbol{W}_i \boldsymbol{C} \quad \forall i \leqslant n \end{cases} . \tag{21}$$

Note that in the PARAFAC2 model, factors $\boldsymbol{A}_i$ are all equal, as well as factors $\boldsymbol{B}_i$, and matrices $\boldsymbol{W}_i$ are left-orthogonal to ensure shared covariance over all factors $\boldsymbol{C}_i$.

## ALGORITHMS

### *Noise covariance and Preprocessing*

Before discussing some workhorse algorithms for constrained CP decomposition, we show that linear constraints can be handled through manipulations of the noise distribution. To describe the distribution of a random multiway array, we use the array normal law introduced in [54].

*Definition:* Let $\boldsymbol{\mathcal{T}}$ be a multivariate random variable in $\mathbb{R}^{n_1 \times \cdots \times n_N}$. We say that $\boldsymbol{\mathcal{T}}$ follows an array normal law of mean $\boldsymbol{\mathcal{M}}$ and with tensor covariance $\boldsymbol{\Gamma} = \bigotimes_{i=1}^{N} \boldsymbol{\Sigma}_i$ if and only if

$$p\left(\boldsymbol{\mathcal{T}} | \boldsymbol{\mathcal{M}}, \boldsymbol{\Gamma}\right) = \frac{\exp\left(-\frac{\|\boldsymbol{\Gamma}^{-\frac{1}{2}}(\boldsymbol{\mathcal{T}}-\boldsymbol{\mathcal{M}})\|_F^2}{2}\right)}{(2\pi)^{\prod_i \frac{n_i}{2}} |\boldsymbol{\Gamma}|^{\frac{1}{2}}} \tag{22}$$

where $\boldsymbol{\Gamma}^{-\frac{1}{2}} = \bigotimes_{i=1}^{N} \boldsymbol{\Sigma}_i^{-\frac{1}{2}}$ and $\boldsymbol{\Sigma}_i$ are full rank symmetric, and $\|\boldsymbol{\mathcal{T}}\|_F^2 = \sum_{ijk} T_{ijk}^2$ is the squared Frobenius norm. This is denoted in short as $\boldsymbol{\mathcal{T}} \sim \mathcal{AN}\left(\boldsymbol{\mathcal{M}}, \boldsymbol{\Gamma}\right)$.

The array normal law is especially useful when the data tensor is transformed by a multilinear operator. Indeed, suppose $\boldsymbol{\mathcal{T}}$ follows an array normal law of mean $\boldsymbol{\mathcal{M}}$ with diagonal covariance $\bigotimes_i \boldsymbol{I}$, *i.e.* $\boldsymbol{\mathcal{T}} = \boldsymbol{\mathcal{M}} + \boldsymbol{\mathcal{E}}$ with white Gaussian noise $\boldsymbol{\mathcal{E}}$, then given full column rank matrices $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{W}$, $\boldsymbol{\mathcal{G}} = (\boldsymbol{U} \otimes \boldsymbol{V} \otimes \boldsymbol{W})\, \boldsymbol{\mathcal{T}}$ is distributed as

$$\boldsymbol{\mathcal{G}} \sim \mathcal{AN}\left((\boldsymbol{U} \otimes \boldsymbol{V} \otimes \boldsymbol{W})\, \boldsymbol{\mathcal{M}}, \boldsymbol{U}\boldsymbol{U}^T \otimes \boldsymbol{V}\boldsymbol{V}^T \otimes \boldsymbol{W}\boldsymbol{W}^T\right). \tag{23}$$

If $\boldsymbol{U}$ is however not full column rank, then the array normal distribution is degenerate and does not admit a density function, a property inherited from multivariate normal distributions.

This means that an algorithm designed for data corrupted by an additive white Gaussian noise can be used for tensors with known non diagonal but separable covariance, *i.e.* a known covariance as in the above definition. This noisy tensor only needs to be preprocessed with an adequate multilinear operator. On the other hand, when dealing with linear constraints on factors, preprocessing the tensor can allow us to get rid of the constraints *at the cost of correlating the noise*. This means that algorithmically, model (5) with white Gaussian noise can be handled as an unconstrained optimization problem with correlated Gaussian noise; more explicitly, if the mixing matrix $\boldsymbol{W}$ has a left pseudo-inverse $\boldsymbol{W}^\dagger$, the linearly constrained CP decomposition model (5) is equivalent to

$$\begin{cases} \boldsymbol{\mathcal{G}} &= (\boldsymbol{I} \otimes \boldsymbol{I} \otimes \boldsymbol{W}^\dagger)\boldsymbol{\mathcal{T}} \\ &= (\boldsymbol{A} \otimes \boldsymbol{B} \otimes \boldsymbol{C_c})\, \boldsymbol{\mathcal{I}}_R + \boldsymbol{\mathcal{E}}' \\ \boldsymbol{\mathcal{E}}' &\sim \mathcal{AN}\left(0, \boldsymbol{I}_K \otimes \boldsymbol{I}_L \otimes \boldsymbol{W}^\dagger \boldsymbol{W}^{\dagger T}\right) \end{cases} . \tag{24}$$

Preprocessing a data tensor to remove the linear constraint is another way to understand compression. An instance of compression computed using a basis of splines for factor $\boldsymbol{C}$ can be found in chemometrics [24], but only the QR decomposition of this basis was used to preprocess, to avoid to correlate the noise. However, correlating the noise is not an issue since the correlation is known.

*Projected algorithms*

Unconstrained CP decomposition remains a vast field of research in terms of optimization algorithms. Still, popular algorithms all depend on the gradient of a well-chosen cost function. For Gaussian noise with separable covariance $\boldsymbol{\Gamma}$ and constrained factor $\boldsymbol{C}$, a cost function $\gamma$ is derived from (3)

$$\forall \boldsymbol{C} \in \mathcal{S}_c, \quad \gamma\left(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}\right) = \left\| \boldsymbol{\Gamma}^{-\frac{1}{2}} \left( \boldsymbol{\mathcal{T}} - \left(\boldsymbol{A}\otimes\boldsymbol{B}\otimes\boldsymbol{C}\right)\boldsymbol{\mathcal{I}}_R \right) \right\|_F^2, \tag{25}$$

but depending on the exact nature of $\mathcal{S}_c$, its gradient w.r.t. $\boldsymbol{C}$ may not easy to compute in general. However, for all constraint spaces introduced above, it is possible to define a projection operator $\Pi$ mapping $\mathbb{R}^{M\times R}$ to $\mathcal{S}_c$. When the descent direction and a step is given, a projected algorithm updates the current values of the parameters and then projects them onto the constraint space $\mathcal{S}_c$ using $\Pi$. For non-negativity constraints, $\Pi(\boldsymbol{C})$ sets to zero all negative values of $\boldsymbol{C}$ [55]. For sparsity constraints, projecting means thresholding values in the factor until the constraints are satisfied, which is an instance of the well-studied water-filling problem.

Most early algorithms for constrained CP in the literature have used projected gradient as a standard [2]. For completeness, we give below the gradient of (25) with no constraints, which can be set to zero factor-wise to obtain the renowned Alternating Least Squares algorithm. The gradient can also be used as is for projected conjuguate gradient [56] or second order methods [57], [58].

Defining the matricization of an array as in [19] with last mode being the first rolling index, *i.e.* $[\boldsymbol{T}_{(1)}]_{k,\ell M+m} = \boldsymbol{\mathcal{T}}_{k\ell m}$, the gradient of $\gamma$ along the first mode is given by

$$\begin{aligned}\frac{\partial\gamma}{\partial\boldsymbol{A}}\left(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}\right) = \boldsymbol{U}^{-1}\boldsymbol{T}_{(1)}\left(\left(\boldsymbol{V}\boldsymbol{V}^T\right)^{-1}\boldsymbol{B}\odot\left(\boldsymbol{W}\boldsymbol{W}^T\right)^{-1}\boldsymbol{C}\right)\\ -\ \boldsymbol{U}^{-1}\boldsymbol{A}\left(\boldsymbol{B}^T\left(\boldsymbol{V}\boldsymbol{V}^T\right)^{-1}\boldsymbol{B}\boxdot\boldsymbol{C}^T\left(\boldsymbol{W}\boldsymbol{W}^T\right)^{-1}\boldsymbol{C}\right).\end{aligned} \tag{26}$$

The gradients on the other modes are obtained by circulating the variables and covariances since the cost function has the same shape with respect to the three modes when constraints on $\boldsymbol{C}$ are dropped.

The problem with alternating gradient descent and projection steps is that there is no guaranty of convergence to a local minimum. Worse, the algorithm may not converge at all if no critical point of the unconstrained cost function may be reached locally while satisfying the constraints. On the other hand, those algorithms are easy to design if unconstrained algorithms are given. Some results on convergence for more general block coordinate descent methods can be found in [59]–[61]?

*Proximal algorithms*

Recently more complex methods have been proposed to tackle constrained tensor decompositions, namely proximal methods [62]. The latter offer a wide variety of algorithms depending on the specificities of the optimization problem at hand. ADMM [63] was designed and implemented for tensor decompositions under many different constraint types [43], [64]. Although to our knowledge FISTA and Douglas-Rachford algorithms have not been studied in the context of CP decomposition, proximal gradient [63] and accelerated proximal gradient [65] have been studied in the context of non-negative CP decomposition (12). Seemingly alternating least squares methods incorporating proximal sets, called Outer-loop ADMM in [43], is an efficient way to apply proximal methods to constrained CP decompositions. In Sidebar 3, the workhorse Alternating Non-negative Least Square algorithm [55] for non-negative CP decomposition is cast as proximal gradient algorithm with optimal step size.

**Algorithm 1** ANLS / Alternate Outerloop Proximal Gradient algorithm for non-negative CP decomposition.

Given $\boldsymbol{\mathcal{T}}$ and initial factors $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$,
**while** convergence criterion is not met **do**

$$\begin{aligned}\boldsymbol{A} &= \boldsymbol{\mathcal{T}}_{(1)}\left(\boldsymbol{B}\odot\boldsymbol{C}\right)\left(\boldsymbol{B}^{\mathsf{T}}\boldsymbol{B}\boxdot\boldsymbol{C}^{\mathsf{T}}\boldsymbol{C}\right)^{-1}\\ \boldsymbol{B} &= \boldsymbol{\mathcal{T}}_{(2)}\left(\boldsymbol{A}\odot\boldsymbol{C}\right)\left(\boldsymbol{A}^{\mathsf{T}}\boldsymbol{A}\boxdot\boldsymbol{C}^{\mathsf{T}}\boldsymbol{C}\right)^{-1}\\ \boldsymbol{C} &= \left[\boldsymbol{\mathcal{T}}_{(3)}\left(\boldsymbol{A}\odot\boldsymbol{B}\right)\left(\boldsymbol{A}^{\mathsf{T}}\boldsymbol{A}\boxdot\boldsymbol{B}^{\mathsf{T}}\boldsymbol{B}\right)^{-1}\right]^{+}\end{aligned}$$

**end while**

In the context of compression, a linear transformation has to be included in the proximal step, which is known to be difficult if the columns of compression matrix $\boldsymbol{W}$ are not orthogonal. Some works focus on fast decomposition of non-negative tensors using compression [47], [65], [66]. But there does not exist any method exploiting known proximal algorithms that work only in the linearly compressed tensor space. A useful tool in this regard would be recent primal-dual methods from [67], [68], which allow a linear transformation of the variables when computing the proximal operator of the characteristic function $\eta$ of the constraint space.

**Sidebar 3: Proximal gradient for non-negative CP decomposition** The proximal gradient method for solving constrained optimization problems is closely related to traditional projected gradient descent methods. Indeed, it is an iterative algorithm that requires to compute the gradient at any given point of the unconstrained cost function, and then projects the new estimate on the constraint space. However, to ensure both convergence and satisfied constraints, the projection on the constraint space is done using a particular operator called the proximal operator. It is easy to prove that projected gradient and proximal gradient are identical if an orthogonal projector on the constraint space is known.

For the solving the non-negative CPD optimization problem

$$\begin{array}{ll} \underset{\boldsymbol{C}}{\operatorname{argmin}} & \|\boldsymbol{\mathcal{T}} - (\boldsymbol{A} \otimes \boldsymbol{B} \otimes \boldsymbol{C})\, \boldsymbol{\mathcal{I}}_R\|_F^2 \\ \text{s.t.} & \boldsymbol{C} \in \mathbb{R}_+^{K \times R}. \end{array} \tag{27}$$

the proximal gradient can be used in a block-coordinate or ALS spirit by computing the gradient and proximal operators for each factor sequentially. Since (27) is linear with respect to each factor, a gradient method with optimal step is equivalent to the least squares update, so that without projections, estimates of factors $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ are estimated sequentially as in traditional ALS.
That leaves the computation of the proximal operator of the non-negativity constraint on $\boldsymbol{C}$. By definition, the proximal operator $\Pi_\lambda$ in this case is given by

$$\Pi_\lambda\left(\boldsymbol{X}\right) = \underset{\boldsymbol{U}}{\operatorname{argmin}}\ \eta(\boldsymbol{U}) + \lambda \|\boldsymbol{X} - \boldsymbol{U}\|_2^2 \tag{28}$$

where $\eta(\boldsymbol{X})$ is the characteristic function of matrices with non-negative coefficients, i.e.

$$\eta(\boldsymbol{U}) = \begin{cases} 0 & \text{if } U_{ij} \geqslant 0 \ \ \forall U_{ij} \\ +\infty & \text{if there exist } U_{ij} < 0 \end{cases} \tag{29}$$

It can be seen that $\Pi_\lambda(\boldsymbol{X}) = [\boldsymbol{X}]^+$ for all $\boldsymbol{X}, \lambda$, which means that all negative values in least squares estimate of $\boldsymbol{C}$ are set to zero while leaving the other values intact. This shows that an alternate outerloop of proximal gradient amounts to the ANLS from [] recalled in Algorithm 1.

## CONCLUSIONS

We introduced a formalism for constrained tensor Canonical Polyadic Decomposition that sheds light on similarities and differences among the many tensor decomposition models proposed in the recent years. Rank, linear and non-linear constraints were first presented, then more intricate models mixing these constraints were discussed. Finally we surveyed decomposition algorithms taking constraints into account.

Overall, we show how constraints can be used to design new tensor decomposition models relying on multilinearity of data with respect to underlying parameters. In particular, we introduce dictionary-based CP as a way to obtain interpretable factors. Moreover, data fusion is cast in terms of jointly constrained CPDs. We also explain why preprocessing of noisy data can deal with some linear constraints.

Many open questions remain, among which we wish to emphasize the followings. First, what kind of constraints restore existence of a best low rank approximate of a tensor. Second, can we further design fast algorithms for constrained tensor decomposition, e.g. using compression. Last, can constraints be used in a probabilistic context to gain some flexibility on how much they should be satisfied.

## APPENDIX A
## PROJECTED NON-NEGATIVE ALS: SOME PROPERTIES

### A. Introduction

In what follows, for a tensor $\boldsymbol{\mathcal{T}}$ (or a matrix, vector), we write $\boldsymbol{\mathcal{T}} \geq 0$ if all the elements are non-negative and $\boldsymbol{\mathcal{T}} > 0$ if all elements are positive. Let $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{K \times L \times M}$ be a non-negative tensor (*i.e.*, $\mathcal{T} \geq 0$), and

$$\boldsymbol{\mathcal{T}} \approx (\boldsymbol{U} \otimes \boldsymbol{V} \otimes \boldsymbol{W})\, \mathcal{G},$$

be its HOSVD approximation [46] for the multilinear rank $(R_1, R_2, R_3)$, *i.e.*, $\boldsymbol{U} \in \mathbb{R}^{K \times R_1}$, $\boldsymbol{V} \in \mathbb{R}^{L \times R_2}$, $\boldsymbol{W} \in \mathbb{R}^{M \times R_3}$, where the matrices $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{W}$ are composed of the first left singular vectors of the corresponding unfoldings of $\boldsymbol{\mathcal{T}}$.

The projected ALS algorithm [47] involves optimization over $\boldsymbol{A}_c \in \mathbb{R}^{R_1 \times R}$, $\boldsymbol{B}_c \in \mathbb{R}^{R_2 \times R}$, $\boldsymbol{C}_c \in \mathbb{R}^{R_3 \times R}$, $R_k \geqslant R$, such that they satisfy

$$\boldsymbol{U}\boldsymbol{A}_c \geq 0, \boldsymbol{V}\boldsymbol{B}_c \geq 0, \boldsymbol{W}\boldsymbol{C}_c \geq 0, \tag{30}$$

where $\boldsymbol{A}_c, \boldsymbol{B}_c, \boldsymbol{C}_c$ have normalized columns. In what follows, we are going to show that the set defined by (30) is nonempty in most cases.

### B. Perron-Frobenius theorems

The following two results are known [69].

**Theorem 1.** *Let $\boldsymbol{A} \in \mathbb{R}^{n \times n}$, be a non-negative matrix $\boldsymbol{A} \geq 0$, such that $\boldsymbol{A}^n \neq 0$. Then*

- *$\rho(\boldsymbol{A})$ (the spectral radius) is an eigenvalue of $\boldsymbol{A}$;*
- *there exists a non-negative vector $\boldsymbol{v} \geq 0$ such that it is an eigenvector corresponding to $\rho(\boldsymbol{A})$,* i.e. *$\boldsymbol{A}\boldsymbol{v} = \rho(\boldsymbol{A})v$.*

**Theorem 2.** *Let $\boldsymbol{A} \in \mathbb{R}^{n \times n}$, be an eventually positive matrix,* i.e. *$(\boldsymbol{A})^k > 0$ for $k \geqslant k_0$. Then*

- *$\rho(\boldsymbol{A})$ is a simple eigenvalue of $\boldsymbol{A}$, the only eigenvalue with absolute value equal to $\rho(\boldsymbol{A})$;*
- *the eigenvector corresponding to $\rho(\boldsymbol{A})$ can be chosen positive,* i.e. *$\boldsymbol{v} > 0$.*

From these two theorems, we deduce a simple corollary on singular values/singular vectors of non-negative matrices.

**Corollary 2.1.** *Let $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $m \leqslant n$ be a non-negative matrix ($\boldsymbol{A} \geq 0$). Let*

$$\boldsymbol{A} = \sum_{j=1}^{m} \sigma_j \boldsymbol{u}_j \boldsymbol{v}_j^\top, \tag{31}$$

*be an SVD of $\boldsymbol{A}$, $\sigma_1 \geqslant \cdots \geqslant \sigma_m \geqslant 0$. Then the following statements hold true.*

1) *There exists an SVD of $\boldsymbol{A}$ such that $\boldsymbol{u}_1 \geq 0$. In particular, if $\sigma_1 > \sigma_2$, then $\boldsymbol{u}_1 \geq 0$.*

2) *If none of the rows of $\boldsymbol{A}$ is orthogonal to all of the others and none of the columns of $\boldsymbol{A}$ is orthogonal to all of the others, then $\sigma_1 > \sigma_2$ and $\boldsymbol{u}_1 > 0$.*

*Proof.* 1) We have that $\boldsymbol{A}\boldsymbol{A}^\top \geq 0$, and by Theorem 1, there exists a non-negative eigenvector of $\boldsymbol{A}\boldsymbol{A}^\top$ corresponding to its largest eigenvalue.
2) If none of the rows of $\boldsymbol{A}$ is orthogonal to all of the others, then the weighted graph corresponding to $\boldsymbol{A}\boldsymbol{A}^\top$ is connected, hence there exists $k$ such that $(\boldsymbol{A}\boldsymbol{A}^\top)^k > 0$. The case with columns is similar, but we need to consider $(\boldsymbol{A}^\top\boldsymbol{A})^k$.

□

**Corollary 2.2.** *A random*[1] *non-negative matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ has a simple largest singular value, and the corresponding left singular vector $\boldsymbol{u}_1$ is positive.*

### C. Feasibility of the projected NN CP approximation problem

**Proposition 2.1.** *Let $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $m \geqslant n$ be a non-negative matrix, and $\boldsymbol{U} = \begin{bmatrix}\boldsymbol{u}_1 & \cdots & \boldsymbol{u}_R\end{bmatrix} \in \mathbb{R}^{m\times R}$ be the matrix of its first $R$ left singular vectors, $R \leqslant n$, obtained from an SVD as in* (31).

1) *If $\boldsymbol{u}_1 \geq 0$, then the cone*

$$\mathcal{K} := \{\boldsymbol{v} \in \mathbb{R}^R \,|\, \boldsymbol{U}\boldsymbol{v} \geq 0\}$$

*contains at least one nonzero vector.*
2) *If $\boldsymbol{u}_1 > 0$, then the cone $\mathcal{K}$ is* solid *(has nonempty interior),* i.e.*,*

$$\operatorname{span}\mathcal{K} = \mathbb{R}^R.$$

*Proof.* 1) Let $e_1 = \begin{bmatrix}1 & 0 & \cdots & 0\end{bmatrix}^\top$. Since $\boldsymbol{u}_1 \geq 0$, we have that $\boldsymbol{U}e_1 \geq 0$.
2) The dual cone of the cone $\mathcal{K}$ is equal to

$$\mathcal{K}^* = \{\boldsymbol{U}^\top\boldsymbol{\alpha} \,|\, \boldsymbol{\alpha} \in \mathbb{R}^m, \boldsymbol{\alpha} \geq 0\}.$$

Since $\boldsymbol{u}_1 > 0$, we have that

$$\boldsymbol{v} \in \mathcal{K}^*, (\boldsymbol{v})_1 \leqslant 0 \Rightarrow \boldsymbol{\alpha} = 0 \Rightarrow \boldsymbol{v} = 0.$$

Therefore,

$$\mathcal{K}^* \cap -\mathcal{K}^* = \{0\},$$

*i.e.*, the cone $\mathcal{K}^*$ is pointed[2].
As shown, for example in [70, p. 53], a dual cone of a pointed cone is a solid cone.

□

**Corollary 2.3.** *1) For a random non-negative tensor $\boldsymbol{\mathcal{T}}$, the set* (30) *is nonempty.*
2) *Generically, the matrices $\boldsymbol{A}_c, \boldsymbol{B}_c, \boldsymbol{C}_c$ satisfying* (30) *have rank $R$.*

*Proof.* The statements 1 and 2 follow from the corresponding statements of Proposition 2.1, because $\boldsymbol{U}$ (resp. $\boldsymbol{V}$ and $\boldsymbol{W}$) is the matrix of left singular vectors of the first (resp. second and third) unfolding matrix of $\boldsymbol{\mathcal{T}}$. □

[1]It is meant here that all matrix entries are independently drawn according to an absolutely continuous probability distribution. Such an array is referred to as "generic".

[2] Alternatively, $\mathcal{K}$ is a pointed cone if $\mathcal{K}\backslash 0$ lies in an open half space, *i.e.* there exists a hyperplane that intersects $\mathcal{K}$ only at 0.

## APPENDIX B
## A PROJECTED ALGORITHM FOR DICTIONARY-BASED CP DECOMPOSITION

Here we present briefly the algorithm and model that were used to obtain the results presented in Figure 1. Hyperspectral images of the Alps taken over time were used [12]. A dictionary $\boldsymbol{D}$ of spectra that should appear in the hyperspectral images is provided, and we suppose only these spectra are of interest. Thus we want the columns of $\boldsymbol{C}$ to be exactly among the columns of the dictionary $\boldsymbol{D}$, and therefore a constraint $\boldsymbol{C} = \boldsymbol{D}\boldsymbol{S}$ where $\boldsymbol{S}$ has exactly one non zero coefficient in each column is imposed. The dictionary-based CP decomposition as described in (17) for this particular framework leads to the following optimization problem:

$$\underset{\boldsymbol{A},\boldsymbol{B},\boldsymbol{S}\in\{0,1\}^{N\times R},\|\boldsymbol{s}_i\|_0=1}{\operatorname{argmin}} \|\boldsymbol{\mathcal{T}} - (\boldsymbol{A}\otimes\boldsymbol{B}\otimes\boldsymbol{D}\boldsymbol{S})\,\boldsymbol{\mathcal{I}}_R\|_F^2 \quad (32)$$

Note that the sum to one constraint on the abundances $\boldsymbol{B}$ was dropped here for simplicity.

There are many possibilities to tackle this non-convex optimization problem. Since we mean only proof of concept here to encourage further research, we used a simple algorithm based on projected Alternating Least Squares, detailed in Algorithm 2. When optimizing over $\boldsymbol{S}$, the linear system is ill-posed because $\boldsymbol{D}$ is over-complete, *i.e.* the system has more parameters than equations. The sparsity constraint should regularize the linear system, but since we alternate the projection step on the constraint space with the least squares update, a Tikhonov regularization on $\boldsymbol{S}$ was added to regularize the linear system with respect to $\boldsymbol{S}$. Regularization parameter $\lambda$ was chosen by hand. Updates on $\boldsymbol{A}$ and $\boldsymbol{B}$ are computed through plain ALS.

**Algorithm 2** Projected ALS with known Dictionary

Given $\boldsymbol{\mathcal{T}}$ and initial factors $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$,
**while** convergence criterion is not met **do**
**$\boldsymbol{A}$ and $\boldsymbol{B}$ updates:**

$$\boldsymbol{A} = \boldsymbol{\mathcal{T}}_{(1)}\,(\boldsymbol{B}\odot\boldsymbol{C})\left(\boldsymbol{B}^\mathsf{T}\boldsymbol{B}\boxdot\boldsymbol{C}^\mathsf{T}\boldsymbol{C}\right)^{-1}$$
$$\boldsymbol{B} = \boldsymbol{\mathcal{T}}_{(2)}\,(\boldsymbol{A}\odot\boldsymbol{C})\left(\boldsymbol{A}^\mathsf{T}\boldsymbol{A}\boxdot\boldsymbol{C}^\mathsf{T}\boldsymbol{C}\right)^{-1}$$

**$\boldsymbol{S}$ first estimate:**

$$\boldsymbol{D}^\mathsf{T}\boldsymbol{D}\boldsymbol{S}\left(\boldsymbol{A}^\mathsf{T}\boldsymbol{A}\boxdot\boldsymbol{B}^\mathsf{T}\boldsymbol{B}\right) + \lambda\boldsymbol{S} = \boldsymbol{\mathcal{T}}_{(3)}\,(\boldsymbol{A}\odot\boldsymbol{B})$$

solved as a Sylvester equation.

**$\boldsymbol{S}$ projection step:**

$$\forall i < R \;\; s_{ij} = \mathbb{1}_{[\max_j s_{ij}]};\;\; \boldsymbol{C} = \boldsymbol{D}\boldsymbol{S}$$

**end while**

The parameters of the CP decomposition model used for Figure 1 where the following: CP rank $R$ was set to 7, maximum number of iterations was set to $10^3$, initial values for factors were drawn with a zero-mean unit-variance normal distribution coefficient-wise, and $\lambda$ was set to $10^{-3}$. Dictionary

$\boldsymbol{D}$ was normalized columns-wise. Comparison with ANLS is done with the same set of parameters and same initial factors. All data and codes are available online[3].

## REFERENCES


[1] A. Smilde, R. Bro, and P. Geladi, *Multi-Way Analysis*. Chichester UK: Wiley, 2004.

[2] R. Bro, "*Multi-way Analysis in the Food Industry: Models, Algorithms, and Applications*," Ph.D. dissertation, University of Amsterdam, The Netherlands, 1998.

[3] M. Mørup, L. K. Hansen, C. S. Herrmann, J. Parnas, and S. M. Arnfred, "Parallel factor analysis as an exploratory tool for wavelet transformed event-related eeg," *NeuroImage*, vol. 29, no. 3, pp. 938–947, 2006.

[4] B. Hunyadi, D. Camps, L. Sorber, W. Van Paesschen, M. De Vos, S. Van Huffel, and L. De Lathauwer, "Block term decomposition for modelling epileptic seizures," *EURASIP Journal on Advances in Signal Processing*, vol. 2014, no. 1, pp. 1–19, 2014.

[5] A. Cichocki, D. Mandic, L. De Lathauwer, G. Zhou, Q. Zhao, C. Caiafa, and H. A. Phan, "Tensor decompositions for signal processing applications: From two-way to multiway component analysis," *Signal Processing Magazine, IEEE*, vol. 32, no. 2, pp. 145–163, 2015.

[6] N. D. Sidiropoulos, R. Bro, and G. B. Giannakis, "Parallel factor analysis in sensor array processing," *IEEE Trans. Sig. Proc.*, vol. 48, no. 8, pp. 2377–2388, Aug. 2000.

[7] A. L. De Almeida, G. Favier, and J. C. M. Mota, "Constrained tensor modeling approach to blind multiple-antenna cdma schemes," *Signal Processing, IEEE Transactions on*, vol. 56, no. 6, pp. 2417–2428, 2008.

[8] L.-H. Lim, "Foundations of numerical multilinear algebra: decomposition and approximation of tensors," Ph.D. dissertation, Stanford University, 2007.

[9] P. Bürgisser, M. Clausen, and M. A. Shokrollahi, *Algebraic Complexity Theory*. Berlin Heidelberg: Springer, 1997, vol. 315.

[10] P. Mccullagh, *Tensor Methods in Statistics*, ser. Monographs on Statistics and Applied Probability. Chapman and Hall, 1987.

[11] M. Espig, W. Hackbusch, A. Litvinenko, H. G. Matthies, and E. Zander, "Efficient analysis of high dimensional data in tensor formats," in *Sparse Grids and Applications*. Springer, 2012, pp. 31–56.

[12] M. A. Veganzones, J. E. Cohen, R. Cabral Farias, J. Chanussot, and P. Comon, "Nonnegative tensor CP decomposition of hyperspectral data," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 52, pp. 2577–2588, 2016.

[13] J. M. Landsberg and G. Ottaviani, "Equations for secant varieties of veronese and other varieties," *Annali di Matematica Pura ed Applicata*, vol. 192, no. 4, pp. 569–606, 2013.

[14] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.

[15] Y. Qi, P. Comon, and L. H. Lim, "Uniqueness of non-negative tensor approximations," *IEEE Trans. Inf. Theory*, vol. 62, no. 4, pp. 2170–2183, 2016, arXiv:1410.8129.

[16] P. Comon, "Tensors: a brief introduction," *IEEE Sig. Proc. Magazine*, vol. 31, no. 3, pp. 44–53, May 2014.

[17] G. Zhou, A. Cichocki, Q. Zhao, and S. Xie, "Nonnegative matrix and tensor factorizations : An algorithmic perspective," *Signal processing Magazine*, vol. 31, no. 3, pp. 54–65, 2014.

[18] W. Hackbusch, *Tensor Spaces and Numerical Tensor Calculus*, ser. Series in Computational Mathematics. Berlin, Heidelberg: Springer, 2012.

[19] J. E. Cohen, "About notations in multiway array processing," *arXiv preprint arXiv:1511.01306*, 2015.

[20] J. B. Kruskal, "Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics," *Linear algebra and its applications*, vol. 18, no. 2, pp. 95–138, 1977.

[21] I. Domanov and L. De Lathauwer, "On the uniqueness of the canonical polyadic decomposition of third-order tensors—part ii: Uniqueness of the overall decomposition," *SIAM Journal on Matrix Analysis and Applications*, vol. 34, no. 3, pp. 876–903, 2013.

[22] C. R. Rao, *Linear Statistical Inference and its Applications*, ser. Probability and Statistics. Wiley, 1965.

[23] J. D. Carroll, S. Pruzansky, and J. B. Kruskal, "Candelinc: A general approach to multidimensional analysis of many-way arrays with linear constraints on parameters," *Psychometrika*, vol. 45, no. 1, pp. 3–24, 1980.


[3]Codes and data used to produce Figure 1 are available at www.gipsa-lab.fr/∼pierre.comon/TensorPackage.


[24] M. E. Timmerman and H. A. Kiers, "Three-way component analysis with smoothness constraints," *Computational statistics & data analysis*, vol. 40, no. 3, pp. 447–470, 2002.

[25] P. Comon, M. Sorensen, and E. Tsigaridas, "Decomposing tensors with structured matrix factors reduces to rank-1 approximations," in *ICASSP'2010*, Dallas, Mar. 14-19 2010.

[26] M. Boizard, R. Boyer, G. Favier, J. E. Cohen, and P. Comon, "Performance estimation for tensor CP decomposition with structured factors," in *ICASSP*, Brisbane, Australia, Apr. 19-24 2015, pp. 3482–3486.

[27] R. C. Farias, J. E. Cohen, C. Jutten, and P. Comon, "Joint decompositions with flexible couplings," in *Latent Variable Analysis and Signal Separation*. Springer, 2015, pp. 119–126.

[28] R. Cabral-Farias, J. E. Cohen, and P. Comon, "Exploring multimodal data fusion through joint decompositions with flexible couplings," *IEEE Trans. Sig. Proc.*, 2016, submitted. [Online]. Available: http://arxiv.org/abs/1505.07717

[29] I. Markovsky, *Low rank approximation: algorithms, implementation, applications*. Springer Science & Business Media, 2011.

[30] X. Luciani, S. Mounier, H. Paraquetti, R. Redon, Y. Lucas, A. Bois, L. Lacerda, M. Raynaud, and M. Ripert, "Tracing of dissolved organic matter from the sepetiba bay (brazil) by parafac analysis of total luminescence matrices," *Marine Environmental Research*, vol. 65, no. 2, pp. 148–157, 2008.

[31] H. Bharath, D. Sima, N. Sauwen, U. Himmelreich, L. De Lathauwer, and S. Van Huffel, "Tensor based tumor tissue type differentiation using magnetic resonance spectroscopic imaging," in *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE*. IEEE, 2015, pp. 7003–7006.

[32] S. F. V. Nielsen and M. Morup, "Non-negative tensor factorization with missing data for the modeling of gene expressions in the human brain," in *Machine Learning for Signal Processing (MLSP), 2014 IEEE International Workshop on*. IEEE, 2014, pp. 1–6.

[33] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[34] N. Gillis *et al.*, "Nonnegative matrix factorization: Complexity, algorithms and applications," Ph.D. dissertation, UCL, 2011.

[35] L.-H. Lim and P. Comon, "Nonnegative approximations of nonnegative tensors," *Journal of chemometrics*, vol. 23, no. 7-8, pp. 432–441, 2009.

[36] C. F. Caiafa and A. Cichocki, "Multidimensional compressed sensing and their applications," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 3, no. 6, pp. 355–380, 2013.

[37] L.-H. Lim and P. Comon, "Multiarray signal processing: Tensor decomposition meets compressed sensing," *Comptes Rendus Mecanique*, vol. 338, no. 6, pp. 311–320, 2010.

[38] R. Bro, R. A. Harshman, N. D. Sidiropoulos, and M. E. Lundy, "Modeling multi-way data with linearly dependent loadings," *Journal of Chemometrics*, vol. 23, no. 7-8, pp. 324–340, 2009.

[39] L. De Lathauwer, "Decompositions of a higher-order tensor in block terms-part II: definitions and uniqueness," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 1033–1066, 2008.

[40] H. Chen, B. Zheng, and Y. Song, "Comparison of parafac and paralind in modeling three-way fluorescence data array with special linear dependences in three modes: a case study in 2-naphthol," *Journal of Chemometrics*, vol. 25, no. 1, pp. 20–27, 2011.

[41] S. Miron and D. Brie, "Some rank conditions for the identifiability of the sparse Paralind model," in *Latent Variable Analysis and Signal Separation*. Springer, 2015, pp. 41–48.

[42] F. Caland, S. Miron, D. Brie, and C. Mustin, "A blind sparse approach for estimating constraint matrices in paralind data models," in *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*. IEEE, 2012, pp. 839–843.

[43] K. Huang, N. D. Sidiropoulos, and A. P. Liavas, "A flexible and efficient algorithmic framework for constrained matrix and tensor factorization," *arXiv preprint arXiv:1506.04209*, 2015.

[44] S. Sahnoun, E.-H. Djermoune, D. Brie, and P. Comon, "A simultaneous sparse approximation method for multidimensional harmonic retrieval," *arXiv preprint arXiv:1507.02075*, 2015.

[45] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.

[46] L. De Lathauwer and J. Vandewalle, "Dimensionality reduction in higher-order signal processing and rank-$(R_1, R_2, , R_n)$ reduction in multilinear algebra," *Linear Algebra and its Applications*, vol. 391, pp. 31–55, 2004.

[47] J. E. Cohen, R. C. Farias, and P. Comon, "Fast decomposition of large nonnegative tensors," *Signal Processing Letters, IEEE*, vol. 22, no. 7, pp. 862–866, 2015.

[48] R. A. Harshman and M. E. Lundy, "Data preprocessing and the extended PARAFAC model," in *Research methods for multimode data analysis.*, H. G. Law, C. W. Snyder Jr, J. A. Hattie, and R. P. McDonald, Eds. Praeger, 1984, pp. 216–284.

[49] E. Acar, R. Bro, and A. K. Smilde, "Data fusion in metabolomics using coupled matrix and tensor factorizations," *Proceedings of the IEEE*, vol. 103, no. 9, pp. 1602–1620, 2015.

[50] M. Sørensen and L. D. De Lathauwer, "Coupled canonical polyadic decompositions and (coupled) decompositions in multilinear rank-$(L_r, L_r, 1)$ terms—part I: Uniqueness," *SIAM Journal on Matrix Analysis and Applications*, vol. 36, no. 2, pp. 496–522, 2015.

[51] B. Rivet, M. Duda, A. Guérin-Dugué, C. Jutten, and P. Comon, "Multimodal approach to estimate the ocular movements during eeg recordings: a coupled tensor factorization method," in *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE*. IEEE, 2015, pp. 6983–6986.

[52] R. A. Harshman, "Parafac2: Mathematical and technical notes," *UCLA working papers in phonetics*, vol. 22, no. 3044, p. 122215, 1972.

[53] H. A. Kiers, J. M. Ten Berge, and R. Bro, "Parafac2-part i. a direct fitting algorithm for the parafac2 model," *Journal of Chemometrics*, vol. 13, no. 3-4, pp. 275–294, 1999.

[54] P. D. Hoff *et al.*, "Separable covariance arrays via the tucker product, with applications to multivariate relational data," *Bayesian Analysis*, vol. 6, no. 2, pp. 179–196, 2011.

[55] A. Cichocki and S.-i. Amari, *Adaptive blind signal and image processing: learning algorithms and applications*. John Wiley & Sons, 2002, vol. 1.

[56] J.-P. Royer, N. Thirion-Moreau, and P. Comon, "Computing the polyadic decomposition of nonnegative third order tensors," *Signal Processing*, vol. 91, no. 9, pp. 2159–2171, 2011.

[57] L. Sorber, M. Van Barel, and L. De Lathauwer, "Tensorlab v2.0," *Available online, January*, 2014.

[58] P. Comon, X. Luciani, and A. L. De Almeida, "Tensor decompositions, alternating least squares and other tales," *Journal of Chemometrics*, vol. 23, no. 7-8, pp. 393–405, 2009.

[59] H. Attouch, J. Bolte, and B. F. Svaiter, "Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods," *Mathematical Programming*, vol. 137, no. 1-2, pp. 91–129, 2013.

[60] N. Li, S. Kindermann, and C. Navasca, "Some convergence results on the regularized alternating least-squares method for tensor decomposition," *Lin. Algebra Appl.*, vol. 438, no. 2, pp. 796–812, Jan. 2013.

[61] L. Grippo and M. Sciandrone, "On the convergence of the block nonlinear Gauss Seidel method under convex constraints," *Operations Research Letters*, vol. 26, pp. 127–136, 2000.

[62] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-point algorithms for inverse problems in science and engineering*. Springer, 2011, pp. 185–212.

[63] N. Parikh and S. P. Boyd, "Proximal algorithms." *Foundations and Trends in optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[64] H. Becker, L. Albera, P. Comon, R. Gribonval, F. Wendling, and I. Merlet, "Brain source imaging: from sparse to tensor models," *IEEE Sig. Proc. Magazine*, vol. 32, no. 6, pp. 100–112, Nov. 2015.

[65] Y. Zhang, G. Zhou, Q. Zhao, A. Cichocki, and X. Wang, "Fast nonnegative tensor factorization based on accelerated proximal gradient and low-rank approximation," *Neurocomputing*, 2016.

[66] G. Zhou, A. Cichocki, and S. Xie, "Fast nonnegative matrix/tensor factorization based on low-rank approximation," *Signal Processing, IEEE Transactions on*, vol. 60, no. 6, pp. 2928–2940, 2012.

[67] L. Condat, "A generic proximal algorithm for convex optimizationapplication to total variation minimization," *Signal Processing Letters, IEEE*, vol. 21, no. 8, pp. 985–989, 2014.

[68] P. L. Combettes and J.-C. Pesquet, "Primal-dual splitting algorithm for solving inclusions with mixtures of composite, lipschitzian, and parallel-sum type monotone operators," *Set-Valued and variational analysis*, vol. 20, no. 2, pp. 307–330, 2012.

[69] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 1990.

[70] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.